

This Page Is Inserted by IFW Operations  
and is not a part of the Official Record

## **BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning documents *will not* correct images,  
please do not report the images to the  
Image Problem Mailbox.**



## PATENT ABSTRACTS OF JAPAN

(11) Publication number: 07297725 A

(43) Date of publication of application: 10 . 11 . 95

(51) Int. Cl

**H03M 7/30**  
**G10L 7/04**  
**H03H 17/02**  
**H04B 1/66**  
**H04B 14/04**  
**H04N 7/30**

(21) Application number: 06082831

(22) Date of filing: 21 . 04 . 94

(71) Applicant: FUJITSU LTD

(72) Inventor: YAMASHITA KIMIAKI  
HAYASHI HIDEYUKI  
MATSUO NAOJI  
INAMOTO YASUSHI  
MATSUDA KIICHI

## (54) BAND SYNTHESIS FILTER

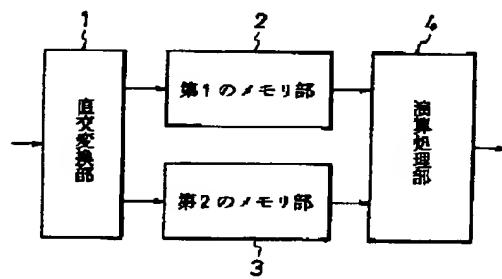
and the memory using quantity are reduced.

## (57) Abstract:

COPYRIGHT: (C)1995,JPO

**PURPOSE:** To reduce the arithmetic quantity of the band synthesis filter by providing an orthogonal transformation section generating a  $2N$ -degree vector from  $N$ -sets of division data, 1st and 2nd memory sections generating a  $2MN$ -degree vector and an arithmetic processing section implementing band synthesis to the filter.

**CONSTITUTION:** An orthogonal transformation section 1 applies orthogonal transformation to  $N$ -sample data to generate a  $2N$ -degree vector. The first half part 0- $N-1$  of the  $2N$ -degree vector and the latter half part  $N-2N-1$  are used for a block unit of the processing and when the first half part is stored in the 1st memory section 2, the latter half part is stored in the 2nd memory section 3. Then an arithmetic processing section 4 implements the processing of band synthesis by using the  $2MN$ -degree vector stored in the 1st memory section 2 storing the first half part. Then the latter half part of the block unit is stored in the 1st memory section 2 and the first half part is stored in the 2nd memory section 3 and the arithmetic processing section 4 implements the processing of band synthesis by using the  $2MN$ -degree vector stored in the 1st memory section 3. Thus, the arithmetic quantity for the filter processing



(19)日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平7-297725

(43)公開日 平成7年(1995)11月10日

(51)Int.Cl. <sup>6</sup>	識別記号	府内整理番号	F I	技術表示箇所
H 03 M 7/30	A	8842-5 J		
G 10 L 7/04	G			
H 03 H 17/02	B	8842-5 J		
H 04 B 1/66				

H 04 N 7/133 Z

審査請求 未請求 請求項の数1 OL (全6頁) 最終頁に統く

(21)出願番号 特願平6-82831

(22)出願日 平成6年(1994)4月21日

(71)出願人 000005223

富士通株式会社

神奈川県川崎市中原区上小田中1015番地

(72)発明者 山下 公彰

福岡県福岡市博多区博多駅前三丁目22番8  
号 富士通九州ディジタル・テクノロジ株  
式会社内

(72)発明者 林 秀之

福岡県福岡市博多区博多駅前三丁目22番8  
号 富士通九州ディジタル・テクノロジ株  
式会社内

(74)代理人 弁理士 柏谷 昭司 (外1名)

最終頁に統く

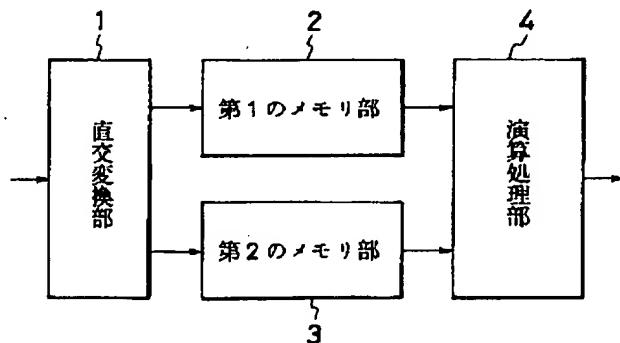
(54)【発明の名称】 帯域合成フィルタ

(57)【要約】

【目的】 帯域分割符号化信号を合成する帯域合成フィルタに関し、演算量を削減する。

【構成】 N帯域に分割されたN個のデータを直交変換して2N次ベクトルを生成する直交変換部1と、この直交変換部1により変換された2N次ベクトルを処理のブロック単位とし、そのブロック単位の前半部と後半部とを交互に切替えてストアし、2MN次ベクトルを生成する第1、第2のメモリ部2、3と、現ブロック単位の前半部をストアしたメモリ部に於ける2MN次のベクトルを用いて帯域合成を行う演算処理部4とを備えている。

本発明の実施例の説明図



## 【特許請求の範囲】

【請求項 1】 N帯域に分割されたデータを入力して帯域合成する帯域合成フィルタに於いて、前記N帯域に分割されたN個のデータを直交変換して2N次のベクトルを生成する直交変換部(1)と、該直交変換部(1)により変換された2N次のベクトルを処理のブロック単位とし、該ブロック単位の前半部と後半部とを交互に切替えてストアしてそれぞれ2MN次のベクトルを生成する第1、第2のメモリ部(2)、(3)と、該第1、第2のメモリ部(2)、(3)の何れか一方の現ブロックの前半部をストアしたメモリ部に於ける2MN次ベクトルを用いて帯域合成を行う演算処理部(4)とを備えたことを特徴とする帯域合成フィルタ。

## 【発明の詳細な説明】

## 【0001】

【産業上の利用分野】 本発明は、帯域分割符号化された信号を合成処理する帯域合成フィルタに関する。音声信号や画像信号等を周波数軸上で帯域分割して処理する各種の方式が知られている。例えば、周波数多重化信号の生成、分離、又は秘話通信の為の周波数スクランブル、又は音声、画像信号のサブバンド符号化等がある。このような帯域分割、合成の手段として、ポリフェーズフィルタバンクが知られており、帯域合成の場合の処理の効率化が要望されている。

## 【0002】

【従来の技術】 音声信号や画像信号等の圧縮符号化方式の一つとして、例えば、国際標準のMPEG (Moving Picture Image Coding Experts Group) が知られている。このMPEGによる音声信号の符号化は、32帯域に分割し、且つ聴覚特性を利用して適応的なビット割当てを行うものである。

【0003】 図4はMPEGオーディオ符号化方式の説明図であり、30は符号化部、40は復号化部、31はマッピング部、32は量子化・符号化部、33は聴覚モデルによる重み付け部、34はビット多重化部、35はビット分離部、36は逆量子化・復号化部、37は逆マッピング部である。

【0004】 入力信号は、32kHz, 41.1kHz, 48kHzの何れかのサンプリング周波数によってサンプリングされてマッピング部31に入力され、この時間領域のサンプルは、64タップのポリフェーズフィルタバンクにより32帯域に分割される。その場合、32サンプルを入力して、480サンプル分のオーバーラップを含めて512サンプルのブロックを形成し、その512サンプルのブロックについて窓関数を乗算した後、前述の64タップのポリフェーズフィルタバンクによる帯域分割処理を行い、32サブバンド出力を得ることができる。

【0005】 量子化・符号化部32は、32帯域毎の最

大値検索により、量子化をフルスケールで行う為のスケールファクターを求め、このスケールファクターによる振幅正規化を、各帯域で12サンプル毎に行う。又スケールファクターの符号化は6ビットのスケールファクターテーブルを用いる。

【0006】 又聴覚モデルによる重み付け部33は、聴覚マスキング効果による重み付けを行うもので、例えば、各サブバンド信号のエネルギーを計算し、この値に最小可聴限の閾値を加算してマスキング閾値とし、このマスキング閾値とサブバンド信号のピークエネルギーとを比較し、マスキング閾値より低い場合はそのサブバンド信号を伝送する必要がないので、量子化・符号化部32に通知する。

【0007】 ビット多重化部34は、サブバンドの符号化データや補助データ等を多重化し、384サンプル単位でフレーム化し、この符号化信号を伝送或いは蓄積する。その場合、例えば、レイヤIのフレーム構成は、16ビットの同期用ビットと、4ビットのビットレート情報と、2ビットのサンプリング周波数情報と、2ビットの符号化モードと、2ビットの著作権の有無情報と、2ビットのエンファシスの有無情報と、8ビットのビット配分情報と、12ビットのスケールファクターと、帯域分割符号化された符号化ビットと、ダミービットとを含むものである。

【0008】 符号化信号の受信側又は再生側に於ける復号化部40は、符号化部30の逆の処理を行うものであり、符号化部30の符号化信号をビット分離部35に於いて補助データ等とサブバンド符号化データとを分離し、逆量子化・復号化部36に於いてサブバンド対応に復号処理する。又逆マッピング部37は、例えば、マッピング部31に於けるポリフェーズフィルタバンクと同様なポリフェーズフィルタバンクを備えて、帯域合成を行い、再生出力信号とするものである (MPEGオーディオ符号化方式は、例えば、1992年10月29日 産業図書株式会社発行、白井克彦監修 海上重之、雁部洋久、池沢斗志、松村俊彦、天野文雄 著「デジタル信号処理の応用」第76頁、第77頁参照)。

【0009】 図5は従来例の帯域合成フィルタバンクのフローチャートであり、前述のMPEGオーディオ符号化方式の逆マッピング部に相当する処理の要部を示す。即ち、ステップ(a)～(f)内で示す演算処理を行うものである。先ず、Nサンプル分のデータS[i] (i=0, 1, 2, ..., N-1) を入力する(a)。又前回の0～4MN-1のベクトルV<sub>i</sub>を2Nサンプル分だけシフトする(b)。又ステップ(a)に於いて入力されたNサンプルの入力信号をモデファイド離散コサイン変換(MDCT)により直交変換して0～2N-1のベクトルを生成し(c)、2Nサンプル分シフトしたベクトルV<sub>i</sub>の先頭に付加して、今回の0～4MN-1のベクトルとする。なお、\*は乗算記号であり、SkはNサ

ンブル  $S[i]$  に相当する。

【0010】次にベクトル  $U_i$  を生成する (d)。このステップは、C 言語表現による場合を示し、直交変換により求めた  $2N$  次ベクトルを、 $2N$  サンプル分シフトした前回のベクトル  $V_i$  の先頭に付加して、 $4MN-1$  サンプル分の新たなベクトル  $V_i$  を生成し、各  $2N$  サンプル分の前半の  $0 \sim N-1$  と後半の  $N \sim 2N-1$  とを選択的に抽出してベクトル  $U_i$  を生成する。

【0011】このベクトル  $U_i$  に窓関数  $D_i$  を乗算してベクトル  $W_i$  を生成する (e)。次に、このベクトル  $W_i$  の疊み込み演算等により帯域合成処理を行う (f)。この合成出力信号  $S_j$  をアナログ信号に変換することにより、入力信号と同じ出力信号が得られる。

【0012】図6は従来例のベクトル生成の説明図であり、図5のステップ (a) に相当する  $0 \sim N-1$  の  $N$  サンプル  $S_i$  を入力し、直交変換処理を行う。それによって、 $0 \sim 2N-1$  の  $2N$  次のベクトルが得られる。又メモリ等に格納された更新前のベクトル  $V_i$  を  $2N$  サンプル分シフトし、その先頭に新たな  $2N$  サンプル分の  $2N$  次ベクトルを付加する。それによって、前回の  $4MN$  サンプルのベクトル  $V_i$  が更新されることになる。

【0013】この更新されたベクトル  $V_i$  に於いて、それぞれの  $2N$  次ベクトルの前半  $0 \sim N-1$  と後半  $N \sim 2N-1$  について、斜線を施して示すように交互に前半と後半とを抽出して、 $0 \sim 2N-1$  のベクトル  $U_i$  を生成する。このベクトル  $U_i$  に窓関数  $D_i$  を乗算してベクトル  $W_i$  を生成し、このベクトル  $W_i$  を基に帯域合成した信号  $S_j$  を得ることができる。

【0014】次に、新たな  $2N$  サンプル分の  $2N$  次ベクトルが直交変換によって得られると、前回の  $4MN$  サンプルのベクトル  $V_i$  を更新し、前回、それぞれの  $2N$  次ベクトルの前半を抽出した場合はその後半を、又後半を抽出した場合はその前半をそれぞれ抽出して、次のベクトル  $U_i$  を生成し、そのベクトル  $U_i$  に窓関数  $D_i$  を乗算してベクトル  $W_i$  を生成し、このベクトル  $W_i$  の疊み込み処理により帯域合成して信号  $S_j$  を得るもので、これを順次繰り返すことになる。

### 【0015】

【発明が解決しようとする課題】帯域分割符号化方式に於ける復号化部は、前述のような帯域合成フィルタ処理を行うものであり、この帯域合成フィルタ処理をプロセッサを用いてリアルタイムで実行する場合、演算量を如何に少なくするかが重要となる。前述の従来例の処理に於いては、 $N$  サンプルのデータを入力する毎に直交変換してベクトル  $V_i$  を更新し、そのベクトル  $V_i$  からベクトル  $U_i$  を生成し、そのベクトル  $U_i$  に対して窓関数  $D_i$  を乗算してベクトル  $W_i$  を生成するもので、ベクトル  $V_i$  からベクトル  $U_i$  を形成する処理は比較的複雑であり、帯域合成フィルタ処理の演算量の増大の一因となっている。本発明は、帯域合成フィルタの演算量を削減す

ることを目的とする。

### 【0016】

【課題を解決するための手段】本発明の帯域合成フィルタは、図1を参照して説明すると、 $N$  帯域に分割された  $N$  個のデータを直交変換して  $2N$  次のベクトルを生成する直交変換部1と、この直交変換部1により変換された  $2N$  次のベクトルを処理のブロック単位とし、このブロック単位の前半部と後半部とを交互に切替えてストアし、それぞれ  $2MN$  次のベクトルを生成する第1、第2のメモリ部2、3と、この第1、第2のメモリ部2、3の何れか一方の現ブロックの前半部をストアしたメモリ部に於ける  $2MN$  次ベクトルを用いて帯域合成を行う演算処理部4とを備えている。

### 【0017】

【作用】直交変換部1は、 $N$  サンプルのデータを直交変換して  $2N$  次ベクトルとし、この  $2N$  次ベクトルの前半部  $0 \sim N-1$  と後半部  $N \sim 2N-1$  とを処理のブロック単位とし、第1のメモリ部2に前半部をストアした時は、後半部を第2のメモリ部3にストアし、且つ前半部をストアした第1のメモリ部2にストアされた  $2MN$  次ベクトルを用いて演算処理部4により帯域合成の処理を行い、次に、第1のメモリ部2にはブロック単位の後半部をストアし、第2のメモリ部3に前半部をストアし、この第2のメモリ部3にストアされた  $2MN$  次ベクトルを用いて演算処理部4により帯域合成の処理を行うものである。

### 【0018】

【実施例】図1に於ける直交変換部1は、離散コサイン変換(DCT)又はモデファイド離散コサイン変換(MDCT)等により、 $N$  サンプルのデータを直交変換して、 $2N$  次ベクトルを生成する。その  $2N$  次ベクトルの前半部  $0 \sim N-1$  と後半部  $N \sim 2N-1$  とを処理のブロック単位とする。

【0019】又第1、第2のメモリ部2、3は、ランダムアクセスメモリ(RAM)やシフトレジスタ等によって構成することができ、直交変換処理毎に、 $N$  次ベクトル分だけシフトして今回のブロック単位の前半部をストアしたメモリ部は、次回はブロック単位の後半部をストアする。従って、第1、第2のメモリ部2、3には、ブロック単位の前半部と後半部とが交互に切替えられてストアされて、 $2MN$  次ベクトルを生成していることになる。

【0020】演算処理部4は、ブロック単位の前半部をストアしたメモリ部の  $2MN$  次ベクトルを用いて帯域合成処理を行うものである。この演算処理部4と直交変換部1とは、プロセッサの演算機能によって実現することができる。

【0021】図2は本発明の実施例のフローチャートであり、 $N$  サンプル分のデータ  $S[i]$  ( $i = 0, 1, 2, \dots, N-1$ ) を入力し①、第1、第2のメモリ部

2, 3にストアされている2MN次のベクトルU<sub>1 i</sub>, U<sub>2 i</sub>をそれぞれNサンプル分シフトする②。

【0022】そして、帯域合成処理に於いて、前回、第1のメモリ部2にストアされたベクトルU<sub>1 i</sub>を用いたか否かを判定し③、このベクトルU<sub>1 i</sub>を用いた場合は、ステップ④に示す直交変換を行い、第2のメモリ部3に2N次ベクトルのブロック単位の前半部をストアし、そのベクトルU<sub>2 i</sub>をベクトルW<sub>i</sub>として窓関数D<sub>i</sub>を乗算し⑤、帯域合成処理を行う⑥。

【0023】又前回、第2のメモリ部3にストアされたベクトルU<sub>2 i</sub>を用いた場合は、ステップ⑥に示す直交変換を行い、第1のメモリ部2に2N次ベクトルのブロック単位の前半部をストアし、そのベクトルU<sub>1 i</sub>をベクトルW<sub>i</sub>として窓関数D<sub>i</sub>を乗算し⑦、帯域合成処理を行う⑧。なお、ステップ④, ⑥は同一の直交変換処理であり、ステップ③の判断結果に対応して直交変換処理結果を選択することができる。

【0024】図3は本発明の実施例のベクトル生成の説明図であり、0～N-1のNサンプル分のデータS<sub>[i]</sub>を入力し、直交変換して2N次ベクトルとし、その2N次ベクトルを処理のブロック単位とし、0～N-1の前半部aと、N～2N-1の後半部bとについて、実線矢印で示すように、前半部aを第2のメモリ部3にストアしてa<sub>n</sub>とし、後半部bを第1のメモリ部2にストアしてb<sub>n</sub>とした場合、第2のメモリ部3には、a<sub>n</sub>, b<sub>n-1</sub>, a<sub>n-2</sub>, …からなるベクトルU<sub>1 i</sub>が生成され、第1のメモリ部2には、b<sub>n</sub>, a<sub>n-1</sub>, b<sub>n-2</sub>, …からなるベクトルU<sub>2 i</sub>が生成される。なお、第1, 第2のメモリ部2, 3は、ブロック単位の前半部a又は後半部bをストアする前に、Nサンプル分のシフトを行うものである。或いは、第1, 第2のメモリ部2, 3をシフトレジスタにより構成し、ブロック単位の前半部a又は後半部bをシフトしながら、第1, 第2のメモリ部2, 3にストアすることができる。

【0025】そして、今回のブロック単位の前半部aをストアした第2のメモリ部3のベクトルU<sub>2 i</sub>を用いて、このベクトルU<sub>2 i</sub>に窓関数D<sub>i</sub>を乗算してベクトルW<sub>i</sub>を生成し、このベクトルW<sub>i</sub>を基に帯域合成する。

【0026】次の直交変換による2N次ベクトルの前半部aを、点線矢印で示すように、第1のメモリ部2にス

トアしてa<sub>n+1</sub>とし、後半部bを、点線矢印で示すように、第2のメモリ部3にストアしてb<sub>n+1</sub>とする。従って、ブロック単位の前半部aをストアした第1のメモリ部2には、a<sub>n+1</sub>, b<sub>n</sub>, a<sub>n-1</sub>, b<sub>n-2</sub>, …からなるベクトルU<sub>1 i</sub>が生成されることになり、今回のブロック単位の前半部aをストアして形成されたベクトルU<sub>1 i</sub>を用い、このベクトルU<sub>1 i</sub>に窓関数D<sub>i</sub>を乗算してベクトルW<sub>i</sub>を生成し、このベクトルW<sub>i</sub>の畳み込み処理によって帯域合成し、信号S<sub>[j]</sub>を出力する。前述の動作を繰り返して帯域分割符号化データの帯域合成を行うことができる。

#### 【0027】

【発明の効果】以上説明したように、本発明は、直交変換部1により変換された2N次のベクトルを処理のブロック単位とし、その前半部を第1のメモリ部2にストアとした時は、後半部を第2のメモリ部3にストアし、次回は、ブロック単位の後半部を第1のメモリ部2にストアし、前半部を第2のメモリ部3にストアするように、交互に切替えてストアし、今回前半部をストアしたメモリ部に於ける2MN次ベクトルを用いて演算処理部4により帯域合成を行うものであり、ベクトルU<sub>1 i</sub>, U<sub>2 i</sub>が形成されるから、従来例のように帯域合成処理の度にベクトルV<sub>i</sub>からベクトルU<sub>i</sub>を生成する処理とメモリ使用とを省略することができる。従って、帯域合成フィルタ処理の演算量とメモリ使用量とを削減できる利点がある。

#### 【図面の簡単な説明】

【図1】本発明の実施例の説明図である。

【図2】本発明の実施例のフローチャートである。

【図3】本発明の実施例のベクトル生成の説明図である。

【図4】MP EGオーディオ符号化方式の説明図である。

【図5】従来例の帯域合成フィルタバンクのフローチャートである。

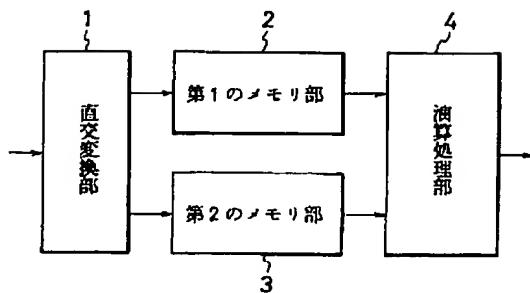
【図6】従来例のベクトル生成の説明図である。

#### 【符号の説明】

- 1 直交変換部
- 2 第1のメモリ部
- 3 第2のメモリ部
- 4 演算処理部

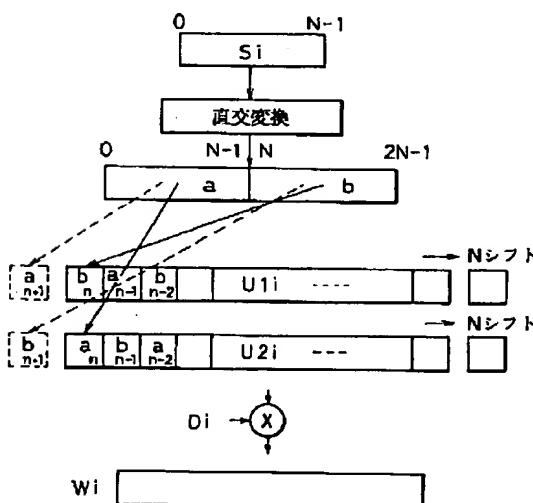
【図1】

本発明の実施例の説明図



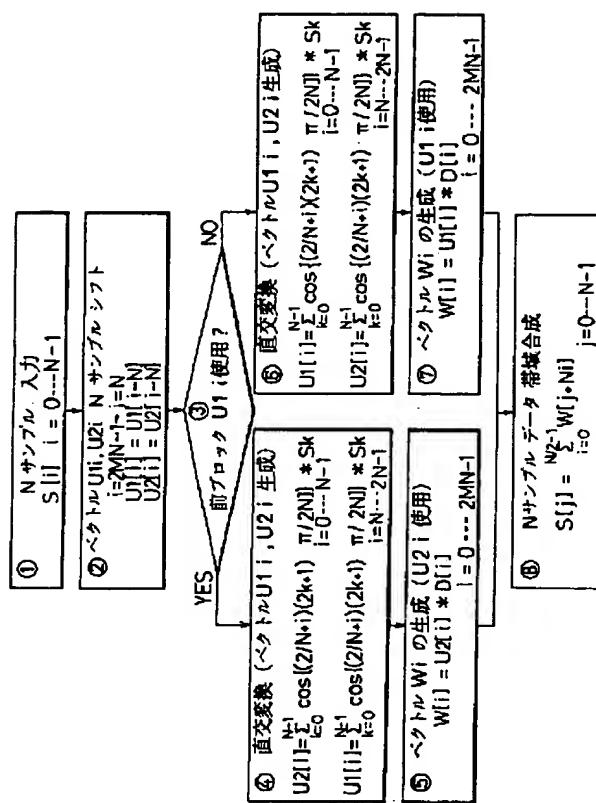
【図3】

本発明の実施例のベクトル生成の説明図



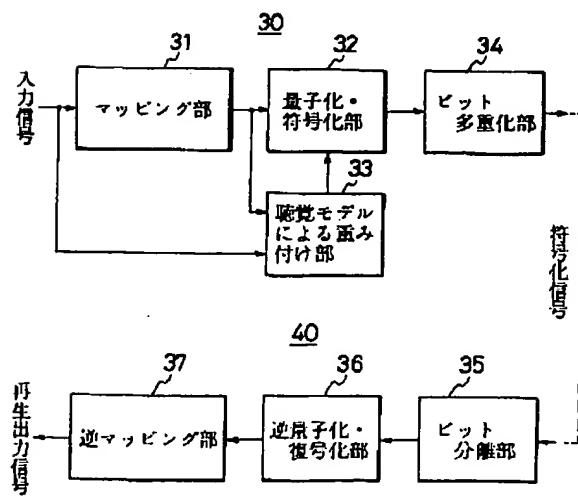
【図2】

本発明の実施例のフローチャート



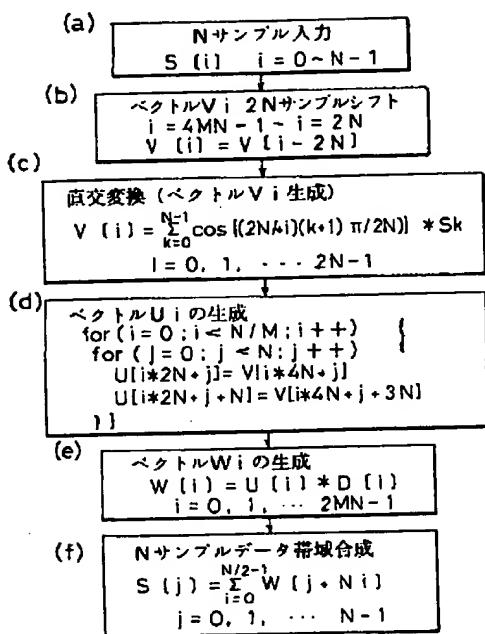
【図4】

MPEGオーディオ符号化方式の説明図



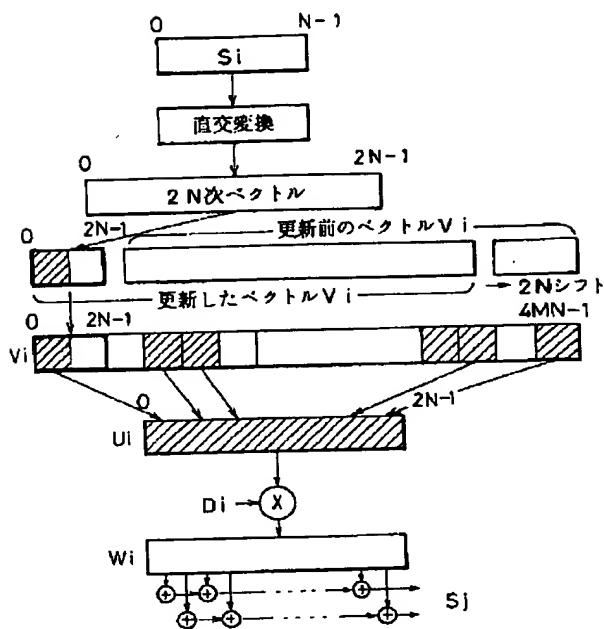
【図5】

従来例の帯域合成フィルタバンクのフローチャート



【図6】

従来例のベクトル生成の説明図



フロントページの続き

(51) Int.C1.6

識別記号 庁内整理番号

F I

技術表示箇所

H 0 4 B 14/04

Z

H 0 4 N 7/30

(72) 発明者 松尾 直司

神奈川県川崎市中原区上小田中1015番地  
富士通株式会社内

(72) 発明者 稲本 康

神奈川県川崎市中原区上小田中1015番地  
富士通株式会社内

(72) 発明者 松田 喜一

神奈川県川崎市中原区上小田中1015番地  
富士通株式会社内